

REMARKS/ARGUMENTS

The Examiner's comments have been carefully reviewed. Claims 1 – 25 are presented for reconsideration and further examination in view of the forgoing amendments and following remarks.

In the outstanding Office Action, the Examiner rejected claims 1 – 12 under 35 U.S.C. 101 as being directed to non-statutory subject matter; rejected claims 1, 3 – 6, 8 – 13, 15 – 18, and 20 – 25 under 35 U.S.C. 102(b) as being anticipated by U.S. Patent No. 6,208,720 to Curtis et al. (hereinafter referred to as "Curtis"); rejected claims 2 and 14 under 35 U.S.C. 103(a) as being unpatentable over Curtis in view of "Enterprise JavaBeans", 2000 by Monson-Haefel (hereinafter referred to as "Monson"); and rejected claims 7 and 19 under 35 U.S.C. 103(a) as being unpatentable over Curtis in view of U.S. Patent No 6,425,039 to Yoshioka et al. (hereinafter referred to as "Yoshioka").

By this Response, applicant has amended the specification to correct a typographical error, amended claims 1, 3 – 5, 13, 15 – 17, 24 and 25, and traverses the above rejections.

It is respectfully submitted that no new matter within the meaning of 35 U.S.C. §132 has been introduced to this application, as the new amendments contain only features that were disclosed in the original specification.

Rejection Under 35 U.S.C. 101

In the outstanding Office Action, the Examiner rejected claims 1 – 12 under 35 U.S.C. 101 as being directed to non-statutory subject matter.

Response

Reconsideration and withdrawal of the rejection are respectfully requested.

Claim 1 has been amended to make it clear that the method is implemented by a computer and further recites that the method relates to a real world transaction that is authorized or blocked depending on whether the situation is logically true or false. Specifically, the claim now adds the feature that the transaction is authorized or blocked in accordance with whether the situation is found to be logically true or false. This is clearly a “real world result” and it is therefore respectfully submitted that these amendments render the subject matter of claim 1 statutory under 35 U.S.C. §101 and that claims 2 to 12 are likewise statutory by virtue of their being dependent on claim 1.

Accordingly, Applicant respectfully requests that the Examiner reconsider and withdraw the rejection of claims 1 – 12 under 35 U.S.C. 101.

Rejection Under 35 U.S.C. 102(b)

In the outstanding Office Action, the Examiner rejected claims 1, 3 – 6, 8 – 13, 15 – 18, and 20 – 25 under 35 U.S.C. 102(b) as being anticipated by Curtis.

Response

Reconsideration and withdrawal of the rejections are respectfully requested.

For a reference to anticipate an invention, all of the elements of that invention must be present in the reference. The test for anticipation under section 102 is whether each and every element as set forth in the claim is found, either expressly or inherently, in a single prior art reference. *Verdegaal Bros. v. Union Oil Co. of California*, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987); MPEP §2131. The identical invention must be shown in as complete detail as is contained in the claim. *Richardson v. Suzuki Motor Co.*, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989); MPEP §2131.

By this Response and Amendment, Applicant respectfully traverses the Examiner's rejection since Curtis fails to disclose, teach, or suggest all of the features of independent claims 1, 13, 24, and 25, and the claims dependent therefrom.

Independent claims 1, 13, 24 and 25 have been amended to more clearly distinguish over Curtis and the objection is respectfully traversed for the following reasons.

Curtis does not teach "a composite threshold "that encapsulates multiple conditions that can be directly compared with a single respective value of a parameter associated with an event and thus obviates the need to compare each of said multiple conditions with the respective value." (Present Application, amended claims 1, 13, 24, and 25, emphasis added). This saves valuable time during real-time processing of real-world events. In order to clarify this and demonstrate why the invention as claimed results in faster processing than Curtis the Examiner is referred to the example given in paragraphs [0018] and [0019] of the present application published as US 2005/0027667, namely:

[0018] In effect such an approach establishes asynchronously a set of binary thresholds that allow synchronous true/false comparison of external "real world" parameters so as to quickly determine whether a situation has occurred or not. In order to understand how such an approach is faster than convention approaches, consider its use in the context of fraud analysis, where the situation relates to the condition that a bank customer is authorized to spend up to \$100 per day on his credit card up to a maximum of \$500 per month. Suppose the customer uses an ATM to withdraw \$50 on the tenth of the month. In a conventional system, the cash withdrawal is first compared with the permissible daily limit. In this case, it is less than the maximum allowed sum. But this on its own does not establish that the transaction is valid since the cumulative cash sum withdrawn prior to the tenth of the month may exceed \$450, in which case the transaction is invalid. Thus, in this very simple example, two independent comparisons must be made.

[0019] In the invention, for the first cash withdrawal during the month, regardless of when it occurs, a single limit of \$100 is set since any financial transaction less than or equal to this sum is valid and may be authorized. Once the customer withdraws any sum, for example, \$50, the threshold is adjusted asynchronously to \$50 since, up until midnight of the same day, this is the maximum allowable limit that may be allowed. After midnight, the threshold reverts to \$100 since the customer has so far spent only \$50 and therefore since the \$500 limit is not exceeded, he may again withdraw \$100 the following day. Suppose that after his first

withdrawal of \$50, he now makes four more \$100 withdrawal during the month. Each of these transactions will be valid since the requested sum is, in each case, less than or equal to the remaining threshold. After the fourth withdrawal, he has thus withdrawn in total \$450 and the threshold is now set asynchronously to \$50. Upon subsequent midnights during the current month, the threshold will remain \$50, and will not be reset to \$100. However, at midnight at the start of the new month, the threshold will again be set to \$100 and the cycle will repeat.

The independent claims recite that following the processing of a current event and **prior** to processing a subsequent event, the current thresholds in the database are updated asynchronously relative to the current event. This prevents processing a subsequent transaction until the database is updated so that all thresholds are current for the new transaction. In contrast to this approach, Curtis *et al.* update their database at some undetermined time **after** a transaction. This means that a system according to the invention will block fraudulent transactions following an event *a priori*, while Curtis will only know that the new transaction was fraudulent *a posteriori*, if the subsequent transaction occurred prior to completion of the process of updating the thresholds.

The independent claims recite that the thresholds are updated asynchronously based on a completed transaction and that the record is locked until such update is performed. No similar mechanism is provided by Curtis.

Curtis updates the rules based on evolving determination of the patterns of events. For example if a customer is used to purchasing only groceries, the system may learn and set rules that associate a given card with typical household consumption transactions. If the card is then used for airline tickets and/or hotel lodging, Curtis's invention may learn, over time, the purchasing habits of the client when traveling and may change or enrich its rules regarding that client.

In contrast thereto, in the present application as claimed the rule is updated not only to reflect a change in pattern but rather to reflect the results of the fact that a transaction occurred. For example, if a threshold is set at \$100 and the customer purchases at \$45 – the threshold will be

immediately set to the remaining amount, i.e. \$55. Moreover, in the invention, once a single authorized transaction is performed abroad, the system may immediately change the rules for the very next transaction to reflect the purchase habits normally associated with traveling.

Claim 1 (as well as the remaining independent claims) recites that prior to processing a subsequent event the current thresholds in the database are updated asynchronously relative to the current event. What this means is that, after a current event is processed and before another event can be processed, all thresholds are re-calculated taking the current event into account. This is done *asynchronously* relative to the current event in that it can be done any time after the current event is processed and before a subsequent event is processed. This asynchronous processing is thus distinct from the initial processing of the current event to determine whether the transaction may be approved or must be blocked, which is signaled by the current event and is therefore done in *synchrony* with the current event. To ensure that the re-calculation of the thresholds taking the current event into account is completed before the subsequent transaction is processed, subsequent events are blocked from further processing until the thresholds are updated. This is quite different from Curtis which does not teach a method for blocking transactions for the given customer until the thresholds are updated.

It is therefore respectfully submitted that claims 1, 13, 24 and 25 are patentable over Curtis.

It is also respectfully submitted that claims 3-6, 8-12, 15-18 and 20-23 are patentable over Curtis by virtue of their being dependent on allowable base claims. In addition, the following comments are offered regarding the distinction of specific ones of these claims over Curtis.

Claim 3 recites that successive thresholds are compared according to a predetermined hierarchy so parameters are processed in progressively decreasing orders of importance. This is known *per se*. The novel aspect is the dynamic nature of the hierarchy. As a prerequisite condition

for speedy execution, the asynchronous processing preparing all the rules and thresholds for the next transaction, determines also which parameters will be evaluated and in which order. So, for example, the parameters may include an initial boundary check as described on page 15, line 3, which is followed by geographic parameters that are constantly updated by the asynchronous processor. In the invention, it is possible that an event will be subjected to evaluation of certain parameters and thresholds, while the subsequent event for the same client will be subjected to a different set of rules, or the same rules in a different order. So, in the invention the hierarchy is updated prior to processing a subsequent event and this distinction has been introduced into claim 3.

Claim 6 recites that at least one parameter relates to a location from which a transaction is performed and the corresponding boundary threshold is a composite threshold that relates to a geographical boundary within which the transaction may be authorized. It is thus noted that this claim refers specifically to the composite threshold, which is not disclosed by Curtis. It is precisely the provision of a composite threshold that allows the present invention to process large quantities of data in real time in a manner that is impossible by Curtis. This is explained in paragraph [0045] of the present application:

[0045] If the transaction does not pass this broad (positive) boundary check, the current transaction is then tested against the conditions set previously for the next transaction, by the asynchronous processing that followed the previous transaction. Transaction-specific evaluation is now invoked. Such evaluation is done against boundary conditions such as the geographical location of the client when the transaction is initiated. For example, if the client performed a transaction on a particular day in London, then it may be asserted that a subsequent transaction carried out within 30 minutes must be somewhere also in London. If, in fact, a transaction is attempted 15 minutes later from New York it can immediately be identified as fraudulent, as explained in greater detail below with particular reference to FIG. 4. So the location of a valid transaction may serve to define a boundary condition that varies with time and may be updated asynchronously. Likewise, transaction-specific boundary conditions may include the maximum amount that may be withdrawn, as described above in the discussion of the composite conditions. Transactions that fail these transaction-specific tests are rejected and a suitable response is generated and conveyed by the synchronous processor 24.

Thus in claim 6, the boundary threshold is a composite threshold that allows determination that a transaction must be blocked by direct comparison with a current event. This is not possible in Curtis, where geographic boundaries require computation of so-called velocity checks. At col. 21, to which the Examiner refers, Curtis maintains a database of longitudes and latitudes for each country and stores times of successive transactions as well as their respective locations. He then computes whether the distance between successive locations is commensurate with the time interval between the two events. This requires computation of the time interval [col. 21, lines 24-25] and comparing with the computed time to travel between the start and end locations at a given maximum travel speed [col. 21, lines 11-12]. It is thus clear that Curtis requires multiple computations: (i) distance between successive events; (ii) time between successive events; (iii) distance that can reasonably be traveled in this time; and (iv) comparison of the distances computed in (i) and (iii). As opposed to Curtis, in claim 6 the boundary threshold is a *composite* threshold that is dynamically updated after a transaction is performed in response to triggers generated by a real-time clock, to establish a global boundary parameter that permits a single comparison. Thus the invention assigns to the next transaction a geographic boundary – such as city, country, continent – that is dynamically updated and can be processed and evaluated significantly faster than the multiple computations required by Curtis.

The same conclusion emerges also from col. 16, lines 50-58 of Curtis, which states:

6) geographic velocity: simultaneous calls over a distance. Rule will provide a minimum time needed between calls, based on physical distance between points of call origination. For example, if a call made with a certain calling card in a first city is placed less than N hours after another call made with the same calling card in a second city, and the second city is more than N hours in traveling time from the first city, then add both events to list that will be counted.

Thus on receiving a second call from a second city, it is necessary for Curtis to calculate the distance between the two cities and determine whether it is more than N hours in traveling time from the first city. This is clearly not amenable to comparison with a single composite boundary threshold.

The patentability of Claim 7 over Curtis is likewise particularly manifest with regard to the use of composite thresholds, which are not disclosed by Curtis. Thus, considering the example, in para. [0019] regarding a bank customer who is authorized to spend up to \$100 per day on his credit card up to a maximum of \$500 per month:

[0019] In the invention, for the first cash withdrawal during the month, regardless of when it occurs, a single limit of \$100 is set since any financial transaction less than or equal to this sum is valid and may be authorized. Once the customer withdraws any sum, for example, \$50, the threshold is adjusted asynchronously to \$50 since, up until midnight of the same day, this is the maximum allowable limit that may be allowed. After midnight, the threshold reverts to \$100 since the customer has so far spent only \$50 and therefore since the \$500 limit is not exceeded, he may again withdraw \$100 the following day. Suppose that after his first withdrawal of \$50, he now makes four more \$100 withdrawal during the month. Each of these transactions will be valid since the requested sum is, in each case, less than or equal to the remaining threshold. After the fourth withdrawal, he has thus withdrawn in total \$450 and the threshold is now set asynchronously to \$50. Upon subsequent midnights during the current month, the threshold will remain \$50, and will not be reset to \$100. However, at midnight at the start of the new month, the threshold will again be set to \$100 and the cycle will repeat.

It emerges from the foregoing that the composite threshold is changed dynamically in response to triggers generated by a real-time clock, which is set or otherwise modified in response to events. It is important to appreciate that, in the above example, after the customer withdraws the first \$50 the threshold is adjusted asynchronously to \$50 and will remain at this value up until midnight of the same day. This threshold is a composite threshold that has no other conditions attached to it: so if the customer subsequently attempts to withdraw \$75, i.e. exceeding the current composite threshold, the transaction will be blocked. On the other hand, after midnight the composite threshold will be re-set to \$100 (providing that the monthly limit is not exceeded). So, within the context of

claim 7, the composite threshold is dynamically recalculated in response to triggers generated by a real-time clock. Since Curtis does not teach the use of composite thresholds, it is clear that claim 7 is patentable over Curtis. Moreover, Curtis relates only to triggers associated with event records that result from client transactions, and does not teach updating the rules or thresholds based on triggers associated with a real time clock.

The Examiner has rejected claim 12 in view of Curtis col. 20, lines 47-50, which describes:

A simultaneous international (SI) alarm is generated when X number of completed international calls using the same authcode/BTN overlap in time by at least N minutes within a sliding window of time T. The X number of international calls is specified by product. The sliding window of time T within which simultaneity is checked can not exceed the purge time for normalized events. An international call is determined from the international indicator in the normalized event.

Even accepting that “network event records” may be regarded as time histories, it is difficult to understand on what basis the Examiner asserts that the sliding time windows teaches that the boundaries of the time histories vary from *client to client* randomly or arbitrarily as claimed in claim 7. Thus, it appears that in Curtis the time window is fixed for all customers handling of all customers is the same. It is clear from the above that the number of transactions is specified by product. It also emerges from col. 16, lines 60-65 of Curtis, there one sliding time window is specified for each rule. Therefore, it appears that in Curtis the sliding window is determined according to a product-related rule and not to the client and we respectfully differ with the examiner’s interpretation that Curtis teaches time boundaries that vary from client to client.

This distinction is rendered even clearer from the following example based on the use of random or arbitrary boundaries according to the invention as claimed in claim 12. Assume that the set of rules and thresholds for all customers includes the following common rules:

1. Maximum customer-specific amount per calendar day;

2. Maximum customer-specific amount per sliding 24 hour time window;
3. Maximum customer-specific amount per sliding 1-hour time window;

Claim 12 is directed to an approach whereby the time windows in which certain transactions may be next allowed are concealed. Thus, according to the invention as claimed in claim 12, the following arbitrary and random time windows are possible:

- a) For each client the fixed calendar day may start at a different arbitrary fixed hour (say 23:05 rather than 24:00 for client A, and 00:37 for client B)
- b) Same as “a” above but the hour when the calendar day starts is periodically set and changed randomly or arbitrarily for each customer. (Say, for client A 24:00 on January 1 and 00:10 on January 2, and for client B 00:37 on January 1, and 00:15 on January 2).
- c) For each client the length of the 24 hour sliding time window may be arbitrarily different, where for some client it may be only 23 hours long, while for others it is 25-hour-and-5-minutes long.
- d) Same as “c” above for the 1-hour time window (say, varying between 51 minutes and 67 minutes).
- e) Same as “c” and “d” above but the sliding time window is set and changed randomly.

By virtue of varying boundaries of the time histories randomly or arbitrarily from *client* to *client* rather than product rule to product rule, it is much more difficult for a thief to discover the changing length of the time window by trial and error, (for purposes of scheduling his theft attempts to the beginning of the next window), as the resulting denied transaction would cause an alarm.

The system and computer program product claims 13-23, and 24-15 are likewise believed to be patentable for the reasons presented above in respect of the corresponding method claims, *mutatis mutandis*.

Thus, as Curtis does not disclose, teach, or suggest all of the claimed features of the Present Application, applicant submits that Curtis cannot anticipate the presently claimed invention.

Accordingly, Applicant respectfully requests that the Examiner reconsider and withdraw the rejection under 35 U.S.C. 102(b) of independent claims 1, 13, 24, and 25, and of dependent claims 3 – 6, 8 – 12, 15 – 18, and 20 – 23 which depend therefrom.

Rejections Under 35 U.S.C. 103(a)

In the outstanding Office Action, the Examiner rejected claims 2 and 14 under 35 U.S.C. 103(a) as being unpatentable over Curtis in view of Monson, and rejected claims 7 and 19 under 35 U.S.C. 103(a) as being unpatentable over Curtis in view of Yoshioka.

Response

Reconsideration and withdrawal of the rejections are respectfully requested. By this Response, Applicant respectfully traverses the Examiner's rejection.

To establish a *prima facie* case of obviousness, the Examiner must establish: (1) that some suggestion or motivation to modify the references exists; (2) a reasonable expectation of success; and (3) that the prior art references teach or suggest all of the claim limitations. *Amgen, Inc. v. Chugai Pharm. Co.*, 18 USPQ2d 1016, 1023 (Fed. Cir. 1991); *In re Fine*, 5 USPQ2d 1596, 1598 (Fed. Cir. 1988); *In re Wilson*, 165 USPQ 494, 496 (CCPA 1970).

Applicant submits that all of the features of the presently claimed invention are not disclosed, taught or suggested in the cited prior art. The arguments above with respect to Curtis are hereby incorporated by reference.

Rejection under Monson

Exclusive write locks in databases that prevent simultaneous attempts to update identical database records have been known for at least 25 years. Monson describes the application of such locks to Enterprise JavaBeans. As opposed to this, the invention as claimed in claims 2 and 14 applies database locks in a new and unique way.

In Monson, the sequence of events is as follows:

- A “write” or “update” transaction is initiated. All the information for the update is contained in the current transaction.
- The database record (or a certain area of the database) is locked.
- The current transaction continues.
- During the processing of the current transaction all other accesses to the locked record or locked area are blocked.
- The current transaction completes.
- The locked record or area is unlocked and other transactions may proceed.

An important aspect of the above scenario is that the updating of the data is a process contained in a current *transaction*. The concept of transaction is central to Monson and to record locking in database updates in general throughout the known art.

If applied to the scenario described in the present application – such transaction would take place as part of the approval and recording of the client’s financial transaction. The client’s financial transaction will be considered complete only when the database is updated as described above.

In the present application, the blocking of operation is substantially different and new in the following aspects:

- **Blocked Operations:** In Monson (and traditional database locking) the lock optionally prohibits certain read, write, or update operations. Claim 2 of the present application does not relate to the blocking of these operations, but rather to financial transactions such as purchases and withdrawals, which require authorization: other transactions, such as deposits, for instance, may be allowed. Monson does not suggest blocking certain kinds of business processes in the manner claimed in claim 2 of the present application for the purposes of maintaining rules for detection of fraud; and such an application is not an obvious extension of traditional database locking.
- **Determination of duration:** In Monson and in traditional database locking, the record is locked while a “current transaction” takes place. This current transaction is encapsulated in a well defined computer program and process. Once the current transaction is complete the record is unlocked. In the present application, the lock is maintained after the current customer financial transaction is complete. This follows by virtue of the fact that the transaction is effectively complete once it is authorized, while the duration of the subsequent blocking is a function of the time required to perform the necessary asynchronous updates, which is performed off-line later. The unique aspect in the present application is locking the customer record to prevent subsequent transactions until the system determines that it is ready to process new transactions from that customer that require authorization. Moreover, since the blocking referred to in claim 2 takes effect precisely during the time period that the thresholds are being re-calculated, it is clear that the database is not locked for updates as is done in Monson and in traditional database locking, but only in respect of new transactions. Indeed, claims 2 and 14 do not claim blocking database records as taught by Monson, but

rather “blocking response to, or rejecting, subsequent events pending completion of updating the current thresholds in the database”.

It is therefore respectfully submitted that the combination of Curtis and Monson fails to teach each of the claimed features of claims 2 and 14, and as such, the Examiner has failed to make a *prima facie* case of obviousness.

Rejection under Yoshioka

Yoshioka is drawn to a single chip data processor responsive to a plurality of events.

Yoshioka fails to cure the deficiencies of Curtis, as Yoshioka does not disclose, teach, or suggest a composite threshold “that encapsulates multiple conditions that can be directly compared with a single respective value of a parameter associated with an event and thus obviates the need to compare each of said multiple conditions with the respective value” as set forth in independent claims 1 and 13, from which claims 7 and 19 depend, respectively.

It is therefore respectfully submitted that the combination of Curtis and Yoshioka fails to teach each of the claimed features of claims 7 and 19, and as such, the Examiner has failed to make a *prima facie* case of obviousness.

CONCLUSION

In light of the foregoing, Applicant submits that the application is now in condition for allowance.

If the Examiner believes the application is not in condition for allowance, Applicant respectfully requests that the Examiner contact the undersigned attorney if it is believed that such contact will expedite the prosecution of the application.

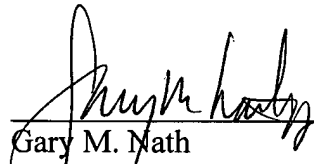
In the event this paper is not timely filed, Applicant petitions for an appropriate extension of time. Please charge any fee deficiency or credit any overpayment to Deposit Account No. 14-0112.

Respectfully submitted,

NATH & ASSOCIATES PLLC

Date: May 3, 2006
NATH & ASSOCIATES PLLC
112 South West Street
Alexandria, VA 22314
(703) 548-6284

By:



Gary M. Nath
Registration No. 26,965
Gregory B. Kang
Registration No. 45,273
Customer No. 20529